UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/783,660	02/14/2001	Peter M. Mansour	SPRODQ1100	9105

25763 7590 02/28/2007
DORSEY & WHITNEY LLP
INTELLECTUAL PROPERTY DEPARTMENT
SUITE 1500
50 SOUTH SIXTH STREET
MINNEAPOLIS, MN 55402-1498

EXAMINER	
LESNIEWSKI, VICTOR D	
ART UNIT	PAPER NUMBER
2152	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	02/28/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	**Application No.** 09/783,660	**Applicant(s)** MANSOUR ET AL.
	Examiner Victor Lesniewski	**Art Unit** 2152

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *28 December 2006*.

2a)☒ This action is **FINAL**. 2b)□ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1,3-8,10-53,55,57-59 and 61-70* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,3-8,10-53,55,57-59 and 61-70* is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)□ The drawing(s) filed on _____ is/are: a)□ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) □ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) □ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application

6) □ Other: _____.

DETAILED ACTION

1. The amendment filed 12/28/2006 has been placed of record in the file.

2. Claims 1, 19, 38, 45, 53, and 59 have been amended.

3. Claims 9 and 56 have been canceled.

4. Claims 1, 3-8, 10-53, 55, 57-59, and 61-70 are now pending.

5. The applicant's arguments with respect to claims 1, 3-8, 10-53, 55, 57-59, and 61-70 have been fully considered but they are not persuasive. A detailed discussion is set forth below.

Response to Amendment

6. Claims have been amended to show the use of a shadow cache at the UI server, this limitation having been previously presented in dependent claims. Since the independent claims have been amended, the rejection of record will be restated below taking into account the amendments.

Claim Rejections - 35 USC § 103

7. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

8. Claims 1, 3-8, 10-53, 55, 57-59, and 61-70 are rejected under 35 U.S.C. 103(a) as being unpatentable over Filepp et al. (U.S. Patent Number 5,347,632), hereinafter referred to as Filepp, in view of Campbell et al. (U.S. Patent Number 6,920,615), hereinafter referred to as Campbell.

9. Filepp disclosed a system that permits users to obtain access via various user interface structures to a large number of applications which include interactive text/graphic sessions. In an analogous art, Campbell disclosed a dynamic web service that generates a user interface for a customer service application based on the customer service, the client device, and the user.

10. Concerning claims 1, 19, 38, 45, 53, and 59, Filepp did not explicitly state formatting characteristics of said intermediate UI based upon a number of device capabilities for said client device. However, this feature was well known in the art as evidenced by Campbell whose system is focused on user interface generation that is based on the customer service application as well as capabilities of the client device. It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to modify the system of Filepp by adding the ability to format characteristics of said intermediate UI based upon a number of device capabilities for said client device as provided by Campbell. Here the combination satisfies the need for an extensible gateway that allows applications to dictate how user interfaces are displayed to the user. See Campbell, column 2, lines 3-10. This rationale also applies to those dependent claims utilizing the same combination.

11. Some claims will be discussed together. Those claims which are essentially the same except that they set forth the claimed invention as a distributed UI architecture, a distributed UI system, or an alternative data processing method are rejected under the same rationale applied to the described claim.

12. Thereby, the combination of Filepp and Campbell discloses:

- <Claims 1 and 53>

A data processing method comprising: generating, with a client device, a particular client-resident intermediate user interface (UI) for a server-based and client-side controlled application according to a UI format determined by a UI server (Filepp, column 10, lines l6-29, where objects in form of packets are sent from server to client, the objects are used for interface generation purposes), including supplementing a skeletal UI stored in a first memory location (skeletal UI is interpreted as incomplete UI; Filepp, column 9, lines 10-34, shows a template UI being populated by different objects, one object is an advertising object) with one or more icons, labels or menu items, or combinations thereof (Filepp, figures 3a, 4c, display types of objects that are available to populate a template page) stored in a second memory location (Filepp, column 5, lines l3-26, where the objects can be stored locally or remotely), wherein the skeletal UI specifies a layout of the client-resident intermediate UI including respective locations of the one or more icons, labels or menu items, or combinations thereof (Filepp, column 10, lines 47-60, where the format of the template is determined using page format objects 502), and wherein the first memory location and the second memory location are situated on said client device (Filepp, column 5, lines 13-26), the skeletal UI and the one or more icons, labels, and menu items being independently updateable from one another (Filepp, column 9, lines 10-34, where individual objects update their respective fields on the page); formatting characteristics of said intermediate UI based upon a number of device capabilities for said client device (Campbell, column 13, lines 33-37 and column 14, lines 48-64); transmitting a number of source data items related to said server-based application from said UI server to said client device (Filepp, column 10, lines l6-29); populating at least one native UI control used by said intermediate UI with said number of source data items (Filepp, column 9, lines 10-34 and column 12, lines 8-17); and maintaining a shadow cache at said UI server, said shadow cache including a list of source data items transmitted from said UI server to said client device (Filepp, column 7, lines 17- 41 and column 5, lines 20-25).

- <Claim 3>

  A method according to claim 1, wherein said at least one native UI control is associated with an operating system for said client device (Filepp, column 4, lines 55-60 and figure 3a, item 290, where the commands are associated with operating system on network device RS 400, the operating system is running each page).

- <Claims 4 and 22>

  A method according to claim 1, further comprising the step of executing, at said UI server, said server-based application to manipulate source data items for presentment at said client device (Filepp, column 5, lines 19-25 and column 7, lines 17-23).

- <Claims 5 and 23>

  A method according to claim 1, further comprising the steps of: generating an action request in response to a manipulation of said intermediate UI by a user of said client device (Filepp, column 7, lines 27-30); and updating said intermediate UI in response to said action request (Filepp, column 7, lines 27-41).

- <Claims 6 and 24>

  A method according to claim 1, further comprising the steps of: performing an offline action by said client device while said client device is disconnected from said UI server (Filepp, column 8, lines 47-61, where the sessions are established via modem, meaning there is no constant connection between client and server; Filepp, column 84, lines 49-60, local storage is checked first for requested object, if not found a remote session is established to the server side for retrieval); subsequently establishing a session between said client device and said UI server (Filepp, column 8, lines 47-61 and column 84, lines 49-60); and thereafter transmitting, from said client device to said UI server, a command indicative of said offline action (Filepp, column 84, lines 49-60, the command is at least in part a GET command to the server side in an attempt to retrieve the objects).

- <Claims 7 and 25>

  A method according to claim 6, further comprising the step of executing said command by said server-based application (Filepp, column 7, lines 25-45, GET command will retrieve objects from the server side).

- <Claims 8 and 26>

  A method according to claim 6, wherein: said offline action modifies at least one of said source data items at said client device (Filepp, column 84, lines 49-60, where modification includes the modification of a display object on the user screen, i.e. user clicks a link on a page, the page is then modified); and said method further comprises the step of updating a corresponding number of source data items maintained by said UI server to reflect the modification of said source data items (Filepp, column 7, lines 18-41, where objects maintained by UI server are sent to the client side to reflect the modification of a user action).

- <Claims 10, 27, 48, 55, and 61>

  A method according to claim 1, further comprising the step of saving said number of source data items in a client cache resident at said client device (Filepp, column 5, lines 20-25).

- <Claims 11 and 28>

  A method according to claim 10, further comprising the step of removing client cache items to accommodate said number of source data items (Filepp, column 6, lines 14-20).

- <Claims 12 and 29>

  A method according to claim 11, wherein said removing step selectively removes said client cache items according to a hierarchical preference scheme (Filepp, column 85, lines 30-40, LRU algorithm is used to maintain cache items).

- <Claims 13 and 32>

  A method according to claim 1, further comprising the steps of: sending a client action command related to said server-based application from said UI server to said client device (Filepp, column 84, lines 50-60); and executing said client action command by said client device (Filepp, column 10, lines 30-57).

- <Claims 14, 41, 51, 58, and 63>

  A method according to claim 1, wherein said number of source data items represent a portion of a larger amount of related data available at said UI server (Filepp, column 7, lines 27-42 and column 5, lines 20-25).

- <Claim 15>

  A method according to claim 14, wherein: said larger amount of related data comprises a list of items; and said number of source data items represents a subset of said list of items (Filepp, column 7, lines 25-42).

- <Claim 16>

  A method according to claim 14, wherein: said larger amount of related data comprises a document (Filepp, column 10, lines 30-41, document is a page); and said number of source data items represents a portion of said document (Filepp, column 9, lines 10-33).

- <Claim 17>

  A method according to claim 14, wherein: said larger amount of related data comprises an image (Filepp, figure 3a, item 255); and said number of source data items represents a portion of said image (Filepp, figure 3a, items 280, 290).

- <Claim 18>

  A method according to claim 14, wherein: said larger amount of related data comprises a body of text (Filepp, figure 3a, item 255); and said number of source data items represents a portion of said body of text (Filepp, figure 3a, item 290).

- <Claim 19>

  A data processing method comprising: defining a user interface (UI) form in response to a number of device capabilities for a client device (Campbell, column 13, lines 33-37 and column 14, lines 48-64), wherein the UI form includes a list of controls and respective locations of the controls as rendered on the client device (Filepp, figure 3a, items 255, 290), the controls being UI objects provided by the client device operating system or other client-resident application (Filepp, column 5, lines 20-25), the UI form and the controls being independently updateable from one another (Filepp, column 9, lines 10-34); storing said UI form locally at said client device (Filepp, column 5, lines 20-25); saving a number of source data items locally at said client device (Filepp, column 5, lines 20-25), said number of source data items being related to a server-based application executed by a UI server (Filepp, column 5, lines 5-25, server can respond to remote requests from the client); maintaining a shadow cache at said UI server, said shadow cache including a list of source data items transmitted from said UI server to said client device (Filepp, column 7, lines 17- 41 and column 5, lines 20-25); and populating said UI form with said number of source data items (Filepp, figure 3a, item 255), wherein said number of source data items comprises a smaller subset than a total number of source data items related to said server-based application, and wherein further subsets of said total number of source data items are downloadable based upon execution of one or more client-side controls (Filepp, column 84, lines 50-60).

- <Claim 20>

  A method according to claim 19, further comprising the step of transmitting said number of source data items from said UI server to said client device (Filepp, column 84, lines 50-60).

- <Claim 21>

  A method according to claim 19, wherein said defining step is performed by said UI server in response to a device identifier obtained from said client device (Filepp, column 24, lines 15-20).

- <Claim 30>

  A method according to claim 27, further comprising the steps of: updating said UI form in response to a manipulation of a display control rendered by said client device (Filepp, column 7, lines 25-42); requesting an additional number of source data items from said UI server if said manipulation of said display control triggers a data request command (Filepp, column 84, lines 49-60); and replacing source data items saved in said client cache with said additional number of source data items (Filepp, column 8, lines 28-40 and column 6, lines 14-18).

- <Claim 31>

  A method according to claim 27, further comprising the steps of: updating said UII form in response to a manipulation of a display control rendered by said client device (Filepp, column 7, lines 25-42); retrieving additional source data items from said client cache in response to said manipulation of said display control (Filepp, column 5, lines 20-25); and displaying said additional source data items in said UI form (Filepp, figure 3 and column. 9, lines 10-32).

- <Claim 33>

  A method according to claim 19, wherein said defining step defines said UI form based upon said server-based application (Filepp, column 5, lines 20-25 and column 7, lines 25-42).

- <Claim 34>

  A method according to claim 19, wherein at least one of the controls is a native UI control stored locally at said client device (Filepp, column 5, lines 20-25).

- <Claim 35>

A method according to claim 19, wherein: said UI server has access to a total number of source data items associated with said UI form (Filepp, column 5, lines 20-25); and said number of source data items saved during said saving step represents a portion of said total number of source data items (Filepp, column 7, lines 27-42 and column 5, lines 20-25).

- <Claim 36>

A method according to claim 35, further comprising the steps of: said UI server receiving a request for additional source data items (Filepp, column 84, lines 50-60); and said UI server transmitting a subsequent portion of said total number of source data items to said client device in response to said request (Filepp, column 84, lines 50-60).

- <Claim 37>

A method according to claim 36, wherein said UI server receives said request from said client device in response to a manipulation of said UI form (Filepp, column 7, lines 25-42).

- <Claim 38>

A data processing method comprising: executing, at a user interface (UI) server, a server-based application configured to manipulate source data items for presentment at a client device (Filepp, column 10, lines 16-29); displaying a particular UI form of a client-resident intermediate UI at said client device according to a UI format determined by a UI server (Filepp, column 10, lines l6-29, where objects in form of packets are sent from server to client, the objects are used for interface generation purposes) and based upon a number device capabilities for said client device (Campbell, column 13, lines 33-37 and column 14, lines 48-64), including the step of supplementing a skeletal UI stored in a first memory location (skeletal UI is interpreted as incomplete UI; Filepp, column 9, lines 10-34, shows a template UI being populated by different objects, one object is an advertising object) with one or more icons, labels or menu items, or combinations thereof (Filepp, figures 3a, 4c, display types of objects that are available to populate a template page) stored in a second memory location (Filepp, column 5, lines 13-26, where the objects can be stored locally or remotely), wherein the skeletal UI specifies a layout of the client-resident intermediate UI including respective locations of the one or more icons, labels or menu items, or combinations thereof (Filepp, column 10, lines 47-60, where the format of the template is determined using page format objects 502), and said UI form being capable of presenting data items to a user of said client device (Filepp, column 9, lines 10-34 and column 12, lines 8-17), wherein the first memory location and the second memory location are situated on said client device (Filepp, column 5, lines 13-26), the skeletal UI and the one or more icons, labels and menu items being independently updateable from one another (Filepp, column 9, lines 10-34, where individual objects update their respective fields on the page); generating a client-side controlled action request in response to a manipulation of said UI form by a user of said client device (Filepp, column 7, lines 27-30); updating said UI form in response to said action request (Filepp, column 7, lines 27-41); and maintaining a shadow cache at said UI server, said shadow cache including a list of source data items transmitted from said UI server to said client device (Filepp, column 7, lines 17- 41 and column 5, lines 20-25).

- <Claim 39>

  A method according to claim 38, further comprising the steps of: sending said action request from said client device to said UI server; and processing said action request by said UI server (Filepp, column 84, lines 50-60).

- <Claim 40>

  A method according to claim 38, further comprising the step of transmitting a number of source data items related to said server-based application from said UI server to said client device, said transmitting step being performed in response to said action request (Filepp, column 84, lines 50-60).

- <Claims 42, 52, and 64>

  A method according to claim 41, further comprising the steps of: requesting, from said UI server, said number of source data items in response to an initial manipulation of said UI form (Filepp, column 84, lines 50-60 and column 5, lines 20-25); and subsequently requesting, from said UI server, an additional number of source data items in response to a further manipulation of said UI form (Filepp, column 84, lines 50-60); wherein said additional number of source data items represent a second portion of said larger amount of related data (Filepp, column 5, lines 20-25, where the data objects are stored remotely on a server).

- <Claim 43>

  A method according to claim 38, further comprising the steps of: said UI server receiving information representing new, deleted, or modified data items (Filepp, column 84, lines 50-60, server receiving requests for new information based on user action/modification on the client side); and said UI server transmitting, to said client device, push data representing said new, deleted, or modified source data items (Filepp, column 84, lines 50-60).

- <Claim 44>

  A method according to claim 43, further comprising the step of said UI server sending, to said client device, a push notification corresponding to said push data (Filepp, column 24, lines 15-30, where the push notification is embedded within the message headers, i.e. the destination ID fields contain notification to the switches as to where to route the corresponding push data).

- <Claim 45>

  A data processing method comprising: generating a user interface (UI) form definition for a server-based application based upon a number of device capabilities for a client device (Campbell, column 13, lines 33-37 and column 14, lines 48-64), wherein the UI form includes a list of controls and respective locations of the controls as rendered on the client device (Filepp, figure 3a, items 255, 290), the controls being UI objects provided by the client device operating system or other client-resident application (Filepp, column 5, lines 20-25), the UI form and the controls being independently updateable from one another (Filepp, column 9, lines 10-34); instructing said client device to render a UI form corresponding to said UI form definition (Filepp, column 10, lines l6-29); rendering said UI form with at least one of the controls associated with the operating system for said client device, wherein the at least one control is a native UI control (Filepp, column 4, lines 55-60 and figure 3a, item 290); transmitting a number of data items from a UI server to said client device (Filepp, column 5, lines 20-25), said number of data items being related to a server-based application (Filepp, column 5, lines 5-25, server can respond to remote requests from the client); maintaining a shadow cache at said UI server, said shadow cache including a list of source data items transmitted from said UI server to said client device (Filepp, column 7, lines 17- 41 and column 5, lines 20-25); and displaying said number of data items in said at least one native UI control (Filepp, figure 3a, item 255), and wherein said number of source data items comprises a smaller subset than a total number of source data items related to said server-based application, and wherein further subsets of said total number of source data items are downloadable based upon execution of one or more client-side controls (Filepp, column 84, lines 50-60).

- <Claim 46>

  A method according to claim 45, further comprising the step of specifying a command script corresponding to a manipulation of a UI control contained in said UI form, said command script being configured for execution by said client device (Filepp, column 40, lines 3-22).

- <Claim 47>

  A method according to claim 46, further comprising the step of executing, by said client device, said command script in response to the manipulation of said UI control at said client device (Filepp, column 40, lines 3-22).

- <Claim 49>

  A method according to claim 48, further comprising the step of retrieving said number of data items from said client cache prior to said displaying step (Filepp, column, lines 5-25).

- <Claim 50>

  A method according to claim 45, further comprising the step of requesting, from said UI server, said number of data items in response to a manipulation of said at least one native UI control (Filepp, column 7, lines 20-42).

- <Claims 57 and 62>

  A distributed UI architecture according to claim 55, wherein said client cache is further configured to store said UI form definition (Filepp, column 5, lines 20-25).

- <Claim 59>

  A distributed user interface (UI) system comprising: a client device having a client processing architecture and a client communication element configured to communicate with a compatible communication element, wherein said client device includes a number of device capabilities related to UI characteristics (Filepp, figure 2, item 405); and a UI server having a server processing architecture and a server communication element configured to communicate with said client communication element (Filepp, figure 2, item 205); said client processing architecture being configured to: transmit a device identifier to said UI server (Filepp, column 24, lines 15-20); generate a UI form in accordance with a UI form definition, wherein the UI form definition includes a list of controls and respective locations of the controls as rendered on the client device (Filepp, figure 3a, items 255, 290), the controls being UI objects provided by the client device operating system or other client-resident application (Filepp, column 5, lines 20-25), the UI form definition and the controls being independently updateable from one another (Filepp, column 9, lines 10-34); and populate at least one of the controls with a number of source data items associated with a server-based application (Filepp, column 5, lines 20-25), wherein the at least on control is a native UI control (Filepp, column 4, lines 55-60 and figure 3a, item 290); said server processing architecture being configured to: receive said device identifier from said client device (Filepp, column 24, lines 15-20); identify said UI form definition in response to service identifier (Filepp, column 24, lines 15-20); and send said number of source data items to said client device for rendering with said UI form (Filepp, column 10, lines 16-29), and wherein said number of source data items comprises a smaller subset than a total number of source data items related to said server-based application, and wherein further subsets of said total number of source data items are downloadable based upon execution of one or more client-side controls (Filepp, column 84, lines 50-60), and wherein said server processing architecture is further configured to generate said UI form definition based upon said number of device capabilities (Campbell, column 13, lines 33-37 and column 14, lines 48-64); and maintain a shadow cache, said shadow cache including a list of source data items transmitted from said UI server to said client device (Filepp, column 7, lines 17- 41 and column 5, lines 20-25).

- <Claims 65, 67, and 69>

  The method of claim 1, wherein said number of source data items comprises a smaller subset than a total number of source data items related to said server-based application, and wherein further subsets of said total number of source data items are downloadable based upon execution of one or more client-side controls (Filepp, column 84, lines 50-60).

- <Claims 66, 68, and 70>

  The method of claim 19, wherein said defined UI form comprises a particular form of a client-resident intermediate UI for a server-based and client-side controlled application according to a UI format determined by a UI server (Filepp, column 10, lines l6-29, where objects in form of packets are sent from server to client, the objects are used for interface generation purposes), and wherein generating the intermediate UI comprises supplementing a skeletal UI stored in a first memory location (skeletal UI is interpreted as incomplete UI; Filepp, column 9, lines 10-34, shows a template UI being populated by different objects, one object is an advertising object) with one or more icons, labels or menu items, or combinations thereof (Filepp, figures 3a, 4c, display types of objects that are available to populate a template page) stored in a second memory location (Filepp, column 5, lines l3-26, where the objects can be stored locally or remotely), wherein the first memory location and the second memory location are situated on said client device (Filepp, column 5, lines 13-26).

Since the combination of Filepp and Campbell discloses all of the above limitations, claims 1, 3-8, 10-53, 55, 57-59, and 61-70 are rejected.

Response to Arguments

13. In the remarks, the applicant has argued:

- <Argument 1>

  Filepp and Campbell are not properly combinable because Filepp teaches away from the claimed invention.

- <Argument 2>

  The combination of Filepp and Campbell does not disclose the features of claim 1 because it does not disclose "formatting characteristics of said intermediate UI based upon a number of device capabilities for said client device" as recited in claim 1.

- <Argument 3>

  The combination of Filepp and Campbell does not disclose the features of claim 1 because it does not disclose "maintaining a shadow cache at said UI server" as recited in claim 1.

- <Argument 4>

  Campbell does not disclose, teach, or suggest formatting, defining, or generating a user interface based upon or in response to a number of device capabilities for a client device, where the user interface, as recited in Applicants' independent claims, contains a user interface form or skeletal user interface and controls, icons, labels, or menu items, each being independently updateable.

14. In response to argument 1, it is maintained that Filepp and Campbell are properly combinable. The applicant states that a "prior art reference must be considered in its entirety,

i.e., as a whole, including portions that would lead away from the claimed invention." It is important to note that this relates to "the claimed invention." While the applicant feels that the present invention is directed toward a thin client architecture and that Filepp teaches away from this, it is noted that a thin client architecture is not a limitation of the claims. The claims simply state an application that is "server-based" and "client-side controlled" and it is maintained that Filepp's programs and applications meet these limitations. If the applicant believes specific details of a thin client architecture would distinguish the present invention from the prior art of record, then these details should be added to the claims.

15. In response to argument 2, the combination of Filepp and Campbell does disclose formatting characteristics of said intermediate UI based upon a number of device capabilities for said client device as recited in claim 1. The applicant mainly argues that Filepp does not teach this limitation. However, it is noted that the rejection of record cites lines to Campbell with regard to this limitation. Here the applicant seems to take the stance that the combination of Filepp and Campbell cannot teach the limitation because Filepp teaches away from it. The applicant argues that Filepp's objects "have a 'uniform, self-defining format,' not a format dependent upon a particular client device." However, no explicit teaching in Filepp could be found that states that these forms cannot be manipulated or that one of ordinary skill in the art would not find an advantage to such a manipulation. It is maintained that one of ordinary skill in the art, taking the teachings of Campbell, would be able to see the advantage of using a format dependent on a particular client device and would have been able to adapt Filepp's system to utilize this advantage. See the previously cited lines to Campbell as well as the discussion of the motivation to combine stated in the rejection above.

16. In response to argument 3, the combination of Filepp and Campbell does disclose maintaining a shadow cache as recited in claim 1. The previous line citations to Filepp, column 7, lines 17-41 and column 5, lines 20-25, show how data (objects and messages) are transmitted from the server to the client in Filepp's system. Filepp clearly utilizes a plurality of cache/concentrator units. See Filepp, figure 2, items 302.

17. In response to argument 4, it is maintained that the combination of Filepp and Campbell teaches the limitations of the applicant's independent claims. Although Campbell may not specifically teach a user interface that contains controls, icons, labels, menu items, etc. as being independently updateable, the combination of Filepp and Campbell does teach this. The applicant is reminded that one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.*, 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986).

18. In addition, the applicant has argued that claims rejected under 35 U.S.C. 103, but not explicitly discussed, are allowable based on the above arguments. Thus, claims disclosing similar limitations to the discussed claims and related dependent claims remain rejected under the same reasoning as presented above.

Conclusion

19. **THIS ACTION IS MADE FINAL.** The applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

20. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Victor Lesniewski whose telephone number is 571-272-3987. The examiner can normally be reached on Monday through Thursday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Bunjob Jaroenchonwanit can be reached on 571-272-3913. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Victor Lesniewski
Patent Examiner
Group Art Unit 2152

BUNJOB JAROENCHONWANIT
SUPERVISORY PATENT EXAMINER